

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
імені ІГОРЯ СІКОРСЬКОГО»

# ІНФОРМАТИКА

## Курсова робота

*Рекомендовано Методичною радою КПІ ім. Ігоря Сікорського як навчальний  
посібник для підготовки бакалаврів, які навчаються  
за спеціальністю 131 – «Прикладна механіка»,  
освітньою програмою «Інжиніринг паковань та пакувального обладнання»*

Київ  
КПІ ім. Ігоря Сікорського  
2019

Інформатика: Курсова робота [Електронний ресурс]: навч. посіб. для підготовки бакалаврів, які навчаються за спеціальністю 131 – «Прикладна механіка», освітньою програмою «Інжиніринг пакування та пакувального обладнання» / КПІ ім. Ігоря Сікорського; уклад.: І. О. Казак. – Електронні текстові дані (1 файл: 3,1 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2019. – 39 с.

*Гриф надано Методичною радою КПІ ім. Ігоря Сікорського (протокол № 8 від 25.04.2019 р.)  
за поданням Вченої ради інженерно-хімічного факультету (протокол № 1 від 28.01.2019 р.)*

Електронне мережне навчальне видання

# ІНФОРМАТИКА

## Курсова робота

Укладач:

*Казак Ірина Олександрівна, канд. пед. наук*

Відповідальний редактор

*Сідоров Д.Е., канд. техн. наук, доц.*

Рецензент:

*Коржик М.В., канд. техн. наук, доц.*

Курсова робота з дисципліни «Інформатика» призначена для формування у студентів основ інформаційних знань та практичних умінь щодо створення програм в Visual Basic for Application (VBA) редактора Excel, що дозволить студентам оптимізувати розрахунки практичних завдань.

Посібник містить теоретичні положення, приклади, контрольні питання і варіанти завдань до виконання курсової роботи з дисципліни «Інформатика». Наприкінці посібника наведено список рекомендованої літератури. Посібник призначений для студентів вищих навчальних закладів, які вивчають дисципліну «Інформатика».

## ЗМІСТ

<b>ВСТУП.....</b>	<b>4</b>
<b>1 МЕТА І ОРГАНІЗАЦІЯ ВИКОНАННЯ КУРСОВОЇ РОБОТИ.....</b>	<b>6</b>
<b>2 ВИМОГИ ДО ОФОРМЛЕННЯ КУРСОВОЇ РОБОТИ.....</b>	<b>7</b>
<b>3 ЗАГАЛЬНІ ПОЛОЖЕННЯ ЩОДО СТВОРЕННЯ І РЕДАГУВАННЯ МАКРОСУ У VBA.....</b>	<b>9</b>
<b>3.1 Створення макросу в VBA.....</b>	<b>10</b>
<b>3.2 Перегляд створеного макросу в VBA.....</b>	<b>11</b>
<b>3.3 Редагування створеного макросу в VBA</b>	<b>12</b>
<b>3.4 Інформація щодо складання блок-схеми для написання програми .....</b>	<b>18</b>
<b>4 ПОРЯДОК ВИКОНАННЯ ЗАВДАННЯ З КУРСОВОЇ РОБОТИ</b>	<b>20</b>
<b>4.1 Створення макросу з побудови графіка функції за варіантом завдання з курсової роботи.....</b>	<b>20</b>
<b>4.2 Модифікація програмного коду макросу з побудови графіка функції за завданням курсової роботи.....</b>	<b>24</b>
<b>4.3 Складання блок-схеми для написання програми в VBA з побудови графіка функції за завданням курсової роботи.....</b>	<b>32</b>
<b>4.4 Створення програми з побудови графіка функції за завданням курсової роботи.....</b>	<b>33</b>
<b>КОНТРОЛЬНІ ПИТАННЯ.....</b>	<b>35</b>
<b>ВАРІАНТИ ЗАВДАНЬ З КУРСОВОЇ РОБОТИ З ДИСЦИПЛІНИ «ІНФОРМАТИКА».....</b>	<b>36</b>
<b>СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ.....</b>	<b>37</b>
<b>СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ.....</b>	<b>37</b>
<b>ДОДАТОК 1. ЗРАЗОК ТИТУЛЬНОГО ЛИСТА.....</b>	<b>38</b>

## ВСТУП

Сучасний рівень технології обробки технічної інформації зумовлює необхідність вивчення студентами вищих навчальних закладів технічних спеціальностей дисципліни «Інформатика».

Відповідно до освітньої програми «Інжиніринг пакування та пакувального обладнання» підготовки бакалавра спеціальності 131 «Прикладна механіка» в процесі засвоєння дисципліни «Інформатика» студенти повинні: оволодіти теоретичними знаннями роботи з інформацією (оформлювати, обробляти, оцінювати, використовувати, редагувати, презентувати); набути навичок практичної роботи з інформацією (виконувати набір текстів, вставки об'єктів, будувати графіки, таблиці та діаграми в редакторах Word, Excel, PowerPoint, а також виконувати обчислення та оцінювати отримані результати, зокрема в програмних середовищах MathCAD, Visual Basic for Application (VBA)) на персональному комп'ютері (ПК). Отримані з даної дисципліни знання є основою для оволодіння дисциплінами: «Інженерні розрахунки на ПЕОМ», «Механіка твердого деформованого тіла», «Процеси і апарати хімічних виробництв», «Сучасні методи розрахунку процесів і апаратів» та іншими дисциплінами спеціальності.

Курсова робота з дисципліни «Інформатика» виконується студентами денної форми навчання на другому році навчання бакалавра спеціальності 131 «Прикладна механіка» освітньої програми «Інжиніринг пакування та пакувального обладнання». Зміст цього навчального посібника відповідає робочій програмі дисципліни «Інформатика».

У курсовій роботі пропонується спосіб автоматичного створення програми за допомогою макрорекордера в VBA редактора Excel на прикладі побудови графіка функції. Таким способом створені програми можна багаторазово використовувати з можливістю їх подальшого редагування, що

дозволить зменшити трудомісткість роботи зі створення програм з типовими діями для студентів, інженерів, дослідників.

Також у курсовій роботі з дисципліни «Інформатика» студентам потрібно модифікувати створений макрорекордером макрос в VBA за завданням та після модифікації створеного макроса студенту необхідно створити програму самостійно за тим же завданням за допомогою застосування потрібних операторів VBA і побудувати для створеної програми блок-схему.

За отриманими результатами виконання завдання курсової роботи трьома способами студентам потрібно порівняти результати розрахунків завдання на збіжність результатів, швидкість і зручність виконання та зробити відповідні висновки у курсовій роботі.

## **1 МЕТА І ОРГАНІЗАЦІЯ ВИКОНАННЯ КУРСОВОЇ РОБОТИ**

Мета курсової роботи з дисципліни «Інформатика» – навчитися студентам створювати програми в Visual Basic for Application (VBA) редактора Excel для вирішення практичних завдань.

Організація виконання курсової роботи включає ряд етапів.

Перш за все, студент повинен ознайомитися з рекомендованою навчальною літературою і уважно вивчити навчальний посібник по виконанню курсової роботи з дисципліни «Інформатика».

Завдання студентам на курсову роботу з дисципліни «Інформатика» видаються викладачем на початку семестру (на момент початку лекцій з дисципліни «Інформатика»). Номер варіанту курсової роботи збігається з номером студента у списку його групи у академічному журналі.

Курсова робота виконується студентами на протязі семестру самостійно і здається на перевірку викладачу за три тижні до завершення семестру. Керівник з курсової роботи, відповідно до встановленого графіка на протязі семестру, здійснює консультації, на яких студенти можуть уточнити змістові незрозумілості і задати питання по виконанню курсової роботи. Після перевірки курсової роботи на відповідність варіанту і правильність виконання завдання студентом викладач призначає дату і час захисту курсової роботи всій групі. Неправильно або не за своїм варіантом виконана курсова робота повертається студенту на доопрацювання з правом захисту після виправлень. Співбесіда викладача зі студентом на захисті дозволяє виявити рівень знань студента з навчального матеріалу курсової роботи, ступінь його самостійності у виконанні курсової роботи. Оцінювання курсової роботи відбувається згідно рейтингової системи оцінювання студентів з даного кредитного модуля.

## **2. ВИМОГИ ДО ОФОРМЛЕННЯ РОЗРАХУНКОВОЇ РОБОТИ**

Курсова робота оформлюється на ПК з використанням текстового редактора Word, а завдання виконується в VBA редактора Excel.

Курсова робота з дисципліни «Інформатика» оформлюється і друкується на аркушах формату A4 з дотриманням таких вимог до технічних документів: шрифт - Times New Roman, розмір шрифту - 14 пунктів, міжрядковий інтервал – 1,5, при форматуванні тексту вирівнювання по ширині, відступ першого рядка абзацу – 1,25 см; поля сторінки: ліве – 30 мм, верхнє, нижнє 20 мм і праве - 10 мм. Сторінки курсової роботи нумеруються посередині знизу аркушів, починають нумерацію сторінок зі вступу (з 3-4 сторінки). Обсяг курсової роботи до 15-20 с.

Курсова робота з дисципліни «Інформатика» виконується і оформлюється за наступною структурою:

**ТИТУЛЬНИЙ ЛИСТ** до курсової роботи з дисципліни «Інформатика», який оформлюється за зразком у додатку 1 даного навчального посібника;

**ЗМІСТ** курсової роботи (сторінки для кожного розділу курсової роботи нумеруються у змісті праворуч у стовпчик);

**ВСТУП** (про актуальність і можливості застосування програм для обробки інформації, зокрема VBA);

**1 УМОВА ЗАВДАННЯ З КУРСОВОЇ РОБОТИ** (згідно варіанта);

**2 ОПИС ДІЙ ЩОДО СТВОРЕННЯ МАКРОСА МАКРОРЕКОРДЕРОМ В VBA З ПОБУДОВИ ГРАФІКА ФУНКЦІЇ** (крім опису наводиться і скрін-шот створеного макроса);

**3 ОПИС ДІЙ ЩОДО МОДИФІКАЦІЇ СТВОРЕННОГО МАКРОСА В VBA** (крім опису наводиться і наводиться скрін-шот модифікованого макроса);

**4 СКЛАДАННЯ БЛОК-СХЕМИ ДЛЯ НАПИСАННЯ ПРОГРАМИ В VBA З ПОБУДОВИ ГРАФІКА ФУНКЦІЇ** (за варіантом завдання до курсової роботи блок-схема до програми оформлюється у Word);

**5 ОПИС ДІЙ ЩОДО НАПИСАННЯ ПРОГРАМИ В VBA З ПОБУДОВИ ГРАФІКА ФУНКЦІЇ** (за завданням до курсової роботи з використанням потрібних операторів і наводиться скрін-шот написаної програми);

**ВИСНОВКИ** до курсової роботи з дисципліни «Інформатика», в яких аналізуються результати виконання завдання трьома способами на збіжність результатів, швидкість і зручність виконання;

**СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ** (оформлюється згідно вимог до оформлення літератури).

Кожен розділ у курсовій роботі оформлюється з нової сторінки і назви розділів оформлюються великими літерами, а пунктів і підпунктів - маленькими.



### **3. ЗАГАЛЬНІ ПОЛОЖЕННЯ ЩОДО СТВОРЕННЯ І РЕДАГУВАННЯ МАКРОСУ У VBA**

На сьогоднішній день, однією з актуальних проблем є трудомісткість створення програм. Одним із способів вирішення цієї проблеми є можливість автоматичного створення програм за допомогою макрорекордера в VBA.

VBA (Visual Basic for Applications) - це діалект мови програмування Visual Basic, який розширює його можливості і призначений для роботи з додатками Microsoft Office і іншими додатками від Microsoft та інших компаній [1, 3, 4].

У більшість програм Microsoft Office вбудовано чудовий засіб, який дозволить вам створювати програми, взагалі нічого не знаючи про програмування. Це засіб називається макрорекордер.

Макрорекордер, як зрозуміло з його назви, - це засіб для запису макросів. Макрос - всього лише ще одна назва для VBA - програми, а макрорекордер - засіб для його автоматичного створення [3].

Повні можливості програмування в Office розкриваються при використанні редактора Visual Basic. Для входу в нього в меню Сервіс | Макрос вибрати «Редактор Visual Basic», або натиснути клавіші <Alt> + <F11>.

У верхньому лівому кутку зазвичай відкрито вікно провідника проекту. Якщо воно випадково було закрито, то викликати його можна, натиснувши клавіші <Ctrl> + <R>.

Самий верхній рівень - це проект (Project), якому відповідає документ Word, робоча книга Excel і т. Д. Проекти VBA виконуються разом з іншими додатками. Додаток, в якому розробляється і виконується проект VBA,

називається основним. Наприклад, можна створити проект VBA, який працює разом з Microsoft Excel. У цьому випадку Excel є основним додатком. Фактично, використовуючи основне додаток, не можна побудувати додаток VBA [3].

VBA має власну середу розробки, називається інтегрованим середовищем розробки або IDE. VBA IDE - це вікно, що містить меню, інші вікна і елементи, що застосовуються при створенні проектів VBA [4]. Нижче меню VBA розташовані програмні модулі. При роботі макрорекордер в Excel - це Module1, куди і записуються всі створювані макрорекордер макроси. Вивести дані програми в VBA можна прямо в клітинку Excel.

### 3.1 Створення макросу у VBA

Щоб створити макрос у вкладках Microsoft Office, наприклад, в Excel за допомогою макрорекордера VBA, необхідно:

1. У меню нової робочої книги Excel знайти у вкладці **«Вид»** → **«Макроси»**, натиснути лівою кнопкою миші на **«Макроси»** і вибрати команду **«Запис макроса»**. У вікні **«Запис макроса»** потрібно написати:

- *ім'я макросу* (ім'я не повинно починатися з цифри, не повинне містити пробіли та символи пунктуації);

- *де зберегти макрос* ( для цього лівою кнопкою миші натиснути на курсор для строки **«Сохранить в»**, вибрати в ній **«Эта книга»** та натиснути кнопку ОК).

2. Після натискання кнопки ОК почнеться запис макросу. В момент включення запису макросу знизу ліворуч з'явиться маленька панель ■, що означає **«Остановить запись»**. Коли всі дії по написанню макросу будуть виконані, то потрібно натиснути знизу ліворуч на маленьку панель ■ - **«Остановить запись»**.

### 3.2 Перегляд створеного макросу у VBA

Для того, щоб переглянути записаний макрос, знизу ліворуч робочої книги Excel натискаємо праву кнопку миші на «Лист 1», обираємо з випадаючого вгору меню «Исходный текст». В результаті цих дій з'являється вікно Microsoft Visual Basic, в якому лівою кнопкою миші двічі натискаємо на папку «Modules» і з'являється «Module1», натискаємо лівою кнопкою миші двічі на «Module1» і бачимо створений макрос під ім'ям макросу, яке було йому надано, наприклад: **Sub Макрос1()**.

Щоб перевірити, як працює записаний макрос, потрібно:

- виділити у робочій книзі Excel створені дані і побудований за ними графік протягуванням лівої клавіші миші, а потім натиснути на клавіатурі клавішу **delete**, щоб видалити з книги Excel виділений фрагмент;
- у меню очищеної робочої книги Excel знайти у вкладці «Вид» → «Макросы», натиснути лівою кнопкою миші на «Макросы», вибрати ім'я створеного макросу, наприклад, **Макрос1()** розташованого в «Эта книга» та натиснути у вікні макросу лівою кнопкою миші на команду «Выполнить». В результаті у очищеній робочій книзі Excel з'являються заново створені макросом дані і побудований за ними графік.

Щоб зберегти заново створений макрос у книзі Excel, потрібно лівою кнопкою миші натиснути у верхньому лівому куту меню Excel на «Кнопка Office», вибрати з випадаючого донизу меню «Сохранить как» → «Книга Excel с поддержкой макросов», у вікні для збереження файлу с макросом записати ім'я файлу з макросом з розширенням **xlsm** і зберегти його у визначеному користувачем місці на комп'ютері.

### 3.3 Редагування створеного макросу у VBA

Для того, щоби внести в макрос будь-які доповнення або виправлення, можна скористатися вікном меню **Вид** → **Макроси** → **Изменить**, або натиснути одночасно клавіші <Alt> + <F11> у робочій книзі Excel.

У верхньому лівому кутку вікна у VBA зазвичай відкрито вікно провідника проекту. Самий верхній рівень – це проект (Project), якому відповідає робоча книга Excel. Нижче розташовані програмні модулі. При роботі макрорекордера в Excel – **Module1**, куди і записуються всі створювані макрорекордером макроси.

Для внесення змін у макросі необхідно знати наступні конструкції VBA.

*До основних конструкцій мови VBA відносяться змінні, константи і службові слова (оператори, процедури, функції).*

*Змінні* - це область оперативної пам'яті, призначена для зберігання будь-якого значення, яке може змінюватися в ході виконання програми.

У VBA передбачено наступні *типи даних змінних*:

- Числові:

- Integer - ціле число від -32 768 до 32 767;
- Long - велике ціле число від -2147483648 до 2147483647;
- Single і Double - значення з плаваючою комою (Double у 2 рази

більше);

- Рядкові (String);

- Логічний (Boolean - може зберігати тільки значення True і False);

- Об'єктний (Object - зберігає посилання на будь-який об'єкт в пам'яті).

Перед роботою зі змінною рекомендується її оголосити оператором Dim.

Приклад роботи зі змінною X цілого типу:

Dim x As Integer

x = x + 10

MsgBox x

Можна оголосити кілька змінних в одному рядку різних типів даних, наприклад, так:

Dim n As Integer, s As String

Рядкові змінні записуються в подвійних лапках. Для злиття строкових змінних використовується оператор &.

Приклад:

Dim Name, Fam, St As String

Name = "Сергій"

Fam = "Петров"

St = "Розробник" & Fam & Name

Деякі найпопулярніші функції, які застосовуються при розробці програм у VBA:

Квадратний корінь - SQR ();

Функція CStr () дозволяє перевести числове значення в строкове;

Abs () - повертає абсолютне значення числа;

Fix () - відкидає дробову частину числа;

sin (), cos (), tan (), atn () - відповідно синус, косинус, тангенс, арктангенс;

log () - натуральний логарифм.

При виконанні математичних розрахунків у VBA використовуються символи математичних операцій:

= привласнення;

- віднімання;

+ додавання;

/ ділення;

\* множення;

^ підведення в ступінь.

Порядок виконання операцій в математичних виразах на мові програмування VBA – загальноприйнятий, як в математиці за пріоритетами математичних операцій, які виконуються в такій послідовності:

1. Математичні функції;
2. Підведення в ступінь;
3. Множення і ділення;
4. Складання і віднімання.

Круглі скобки у математичних виразах ( ) у мові програмування VBA застосовуються для зміни порядку операцій. Спочатку виконуються операції в скобках, потім за межами скобок. Якщо використовують певні математичні дії, то спочатку вони виконуються за пріоритетами математичних операцій у внутрішніх скобках, потім у зовнішніх.

Особливістю запису формул на мові програмування VBA є те, що не можна пропускати у формулах знаки множення при записах математичних виразів у VBA.

Приклад, для математичного запису формули:

$$x = \frac{d}{\sqrt[3]{b^2 - 4ac}}$$

Текст вище представленої формули на мові програмування VBA буде наступний:

$$x=d/(b^2-4*a*c)^(1/3)$$

Для всіх тригонометричних формул (sin (), cos (), tan (), atn () і ін.) одиниці виміру кутів – радіани. Наприклад, порахувати tan (r) якщо r=60 °. Спочатку кут r в градусах перераховують у радіани, а саме, помножують на число  $\pi=3,14$  і поділяють на 180 ° далі розраховують тангенс кута у радіанах у комірку A1:

$$r = 60 * 3.14 / 180$$
$$\text{Range("A1")} = \tan (r)$$

Найпростіший спосіб ввести інформацію від користувача – скористатися вбудованою функцією VBA зі *створення діалогових вікон*

Input Box (). Повний синтаксис *створення діалогових вікон* Input Box () виглядає так:

Input Box ("Запрошення", "заголовок вікна", знач. за замовчуванням)

Приклад:

Dim Name

Name = InputBox ("Введіть ім'я")

З'явиться віконце із запрошенням "Введіть ім'я" та кнопками ОК і Cancel. У більш повній формі, наприклад:

b1 = InputBox ("b1 =", "ширина каналу", 7)

де віконце матиме заголовок "ширина каналу" і введене значення «7».

За замовчанням змінна, введена в віконце, має рядковий тип.

Range - дуже важлива функція. Повертає об'єкт Range, який являє собою діапазон комірок і використовується в Excel практично для будь-яких операцій з комірками.

Вибрати яку-небудь комірку, наприклад, A6: Range ("A6"). Select .

Вибрати діапазон комірок:

Range ("A1: A6"). Select або Range (Cells (1, 1), Cells (6, 1)). Select

Обраний діапазон може бути змінною типу Range, з яким можна надалі оперувати. Наприклад посилання на діапазон комірок з A1 по D10 створюється так:

Dim R As Range

Set R = Worksheets ("Лист1").Range ("A1: D10")

Для вибору одної комірки можна використовувати функцію Cells:

x = Cells (6, 1).

або x = Range ("A6").Value

Найпростіший спосіб вивести інформацію користувачеві - скористатися вбудованою функцією VBA зі створення діалогових вікон MsgBox (Var).

Повний її синтаксис виглядає так:

MsgBox (Змінна, кнопки vbOKCancel, "заголовок вікна")

Приклад:

Result = MsgBox ("Натисніть кнопку", vbOKCancel, "Вікно повідомлення")

Опція Кнопки - наприклад, vbOKCancel видасть 2 кнопки: OK і Cancel. При натисненні OK значенням Result буде обрана кнопка 1, при натисканні Cancel буде обрана кнопка 2.

Або просто: Result = MsgBox (g1, , "g1 =").

Або ще простіше: MsgBox (g1).

Вивести дані можна прямо в комірку. Після того, як потрібна комірка знайдена, для запису в неї використовується властивість Value, наприклад:

x.Value = значення

Якщо комірка вже вибрана, можна записати:

ActiveCell.Value = значення

Або відразу вказати комірку виведення: Cells (6, 1) = значення

Вибрати комірку, зміщену відносно поточної на і вниз та на j вправо:

ActiveCell.Offset (i, j).Select

Перш ніж щось виводити в таблицю, корисно очистити лист, для чого служить команда Cells.Clear .

Formula (або FormulaR1C1) = "= ...." дозволяє записати формулу в комірку.

Приклад: Запис абсолютного посилання на комірку A1 і B1 у формулі зі знаком у імені комірки знака \$:

Range("A3").Formula = "=2\*\$A\$1^2+cos(\$B\$1)"

Або запис відносного посилання на комірку A1 і B1 у формулі:  
"=2\*A1^2+cos(B1)"

Також цей приклад формули можна записати ще двома способами через посилання у формулі на комірку R1C1 зі зміщенням на RC[-1]:

ActiveCell.FormulaR1C1 = "=2\*RC[-1]^2+cos(RC[-1])"



У формулу можуть бути включені і змінні, наприклад змінна **a**. Тоді формулу вище слід переписати так, щоб змінна опинилась поза лапками і з'єднувалась з текстом формули оператором злиття строкових змінних **&**. Наприклад, замість 2 в попередній формулі використаємо змінну **a**, яка буде дорівнювати 2. Тоді другим способом формулу розглянуту вище запишемо:

**ActiveCell.FormulaR1C1 = "=" & a & "\*RC[-1]^2+cos(RC[-1])"**

У таблиці 1 наведені приклади найбільш розповсюджених записів математичних виразів у VBA.

Таблиця 1. Правила запису математичних виразів у VBA

Математичний вираз	Вираз, записаний у VBA
1	2
$5x+12y$	$5 * x + 12 * y$
$\frac{x}{y}$	$x/y$
$y^x$	$y^x$
x	x
$19.55 \cdot 10^{-6}$	$19.55E - 6$ або $19.55D - 6$
$-a^b$	$-a^b$ або $-a^{(b)}$
$a^{-b}$	$a^{(-b)}$
$a^{b+c}$	$a^{(b + c)}$
$10^{-4.7}$	$10^{(-4.7)}$
$10^{4.7}$	$10^{4.7}$
$A \cdot B$	$A * B$
$A \cdot (-B)$	$A * (-B)$
$(a^b)^c$	$(a^b)^c$ або $a^{b^c}$
$\frac{a \cdot b}{c \cdot d}$	$a * b / (c * d)$ або $(a * b) / (c * d)$
$a \cdot 10^4$	$a * 1E4, a * 1D4$ або 10000

Продовження табл. 1

1	2
$ a $	Abs(a)
$\sqrt{b}$	Sqr(b)
$\sqrt[5]{c}$	$c^{1/5}$
$\ln(2b+a)$	Log(2 * b + c)
$e^x$	Exp(x)

### 3.4 Інформація щодо складання блок-схеми для написання програми

Основним підготувальним етапом до етапу розроблення програм за допомогою любых мов програмування являється створення блок-схем за алгоритмом. *Блок-схема алгоритму* – це графічне представлення логічної структури алгоритму, де кожний етап обробки інформації зображується у вигляді геометричних символів (блоків). Графічне зображення блоків і їх призначення, що застосовуються у блок-схемах до програм за алгоритмом, представлено у табл. 2. Існують правила зображення блок-схем алгоритмів. Залежно від послідовності дій розрізняють наступні типи алгоритмів:



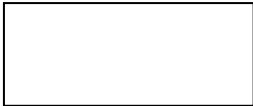
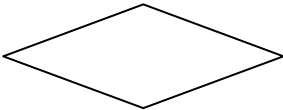

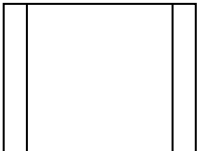
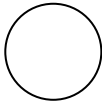
Лінійний алгоритм – це такий алгоритм, в якому дії виконуються в порядку їх запису без пропусків від початку до кінця без розгалуження.

Розгалужений – це такий алгоритм, в якому послідовність дій може змінюватися залежно від виконання чи невиконання деякої умови розгалуження.

Циклічний – це такий алгоритм, в якому деяка група дій виконується багаторазово (має повторювані дії) [7].

Кожен алгоритм має початок та кінець. Кожна команда алгоритму представляється у вигляді геометричних символів, які мають певну конфігурацію, в залежності від характеру дій, що будуть виконуватись.

Таблиця 2 – Графічне зображення і призначення блоків у блок-схемах

Графічне зображення блоку	Призначення
	Початок та кінець алгоритму
	Блок вводу вхідних даних та виведення результату. Цей символ не визначає носія даних
	Виконання однієї або кількох операцій, обробка даних. Всередині фігури записують безпосередньо самі операції.
	Блок умови. В цьому блоці пишеться умова, в залежності від якої вибираються напрямки дії алгоритму. На лініях, що розгалужуються від блоку, пишеться „Так” або „Ні” – залежно від виконання записаної умови.
	Початок і кінець циклу. Межі змінної циклу і крок її збільшення записуються всередині символу початку, а в символі кінця циклу – змінна циклу. Операції, що виконуються всередині циклу, розміщуються між ними.
	Виконання процесу, що складається з операцій, які визначені в іншому місці програми (у підпрограмі, модулі). Всередині символу записується назва процесу і дані, які передані в нього.
	З'єднання. Відображає вихід в частину схеми і вхід з іншої частини цієї схеми. Використовується для обриву лінії та продовження її в іншому місці. Всередині ставиться позначка переходу.

Геометричні символи з'єднуються між собою лініями або лініями зі стрілками, які вказують порядок виконання дій у блок-схемі алгоритму.

## 4 ПОРЯДОК ВИКОНАННЯ ЗАВДАННЯ З КУРСОВОЇ РОБОТИ

### 4.1 Створення макросу з побудови графіка функції за варіантом завдання з курсової роботи

Для створення макросу з побудови графіка функції потрібно:

4.1.1 Увімкнути запис макросу і записати дії з побудови графіка функції  $y(x)$  за варіантом у Excel;

4.1.2 Ввести в комірках A1 та A2 перше та друге значення аргументу;

4.1.3 Розтягнути вміст A1 та A2 на діапазон відображення (наприклад, для 15 значень аргументу функції);

4.1.4 В комірці B1 ввести формулу функції  $y(x)$  у Excel (згідно варіанту завдання до курсової роботи);

4.1.5 Розтягнути вміст комірки B1 на діапазон відображення та представити у курсовій роботі скрін-шот отриманої таблиці даних (Рис.1);

	A	B	C	D	E	F	G	H
1	1	3,7						
2	3	-29,1						
3	5	-3,5						
4	7	138,1						
5	9	453,3						
6	11	999,7						
7	13	1834,9						
8	15	3016,5						
9	17	4602,1						
10	19	6649,3						
11	21	9215,7						
12	23	12358,9						
13	25	16136,5						
14	27	20606,1						
15	29	25825,3						
16								
17								
18								
19								
20								
21								
22								
23								
24								

Рис.1. Створена таблиця даних для побудови графіка функції  $y(x)$  у Excel

4.1.6 Виділити стовпчик В з заповненими комірками, обрати в стрічці Excel у майстрі діаграм вид діаграми (точкова діаграма) і побудувати графік функції  $y(x)$  від заданого діапазону аргументів  $x$ ;

4.1.7 Зупинити запис макросу і представити у курсовій роботі скріншот таблиці даних і побудованого графіку функції  $y(x)$  (Рис.2);

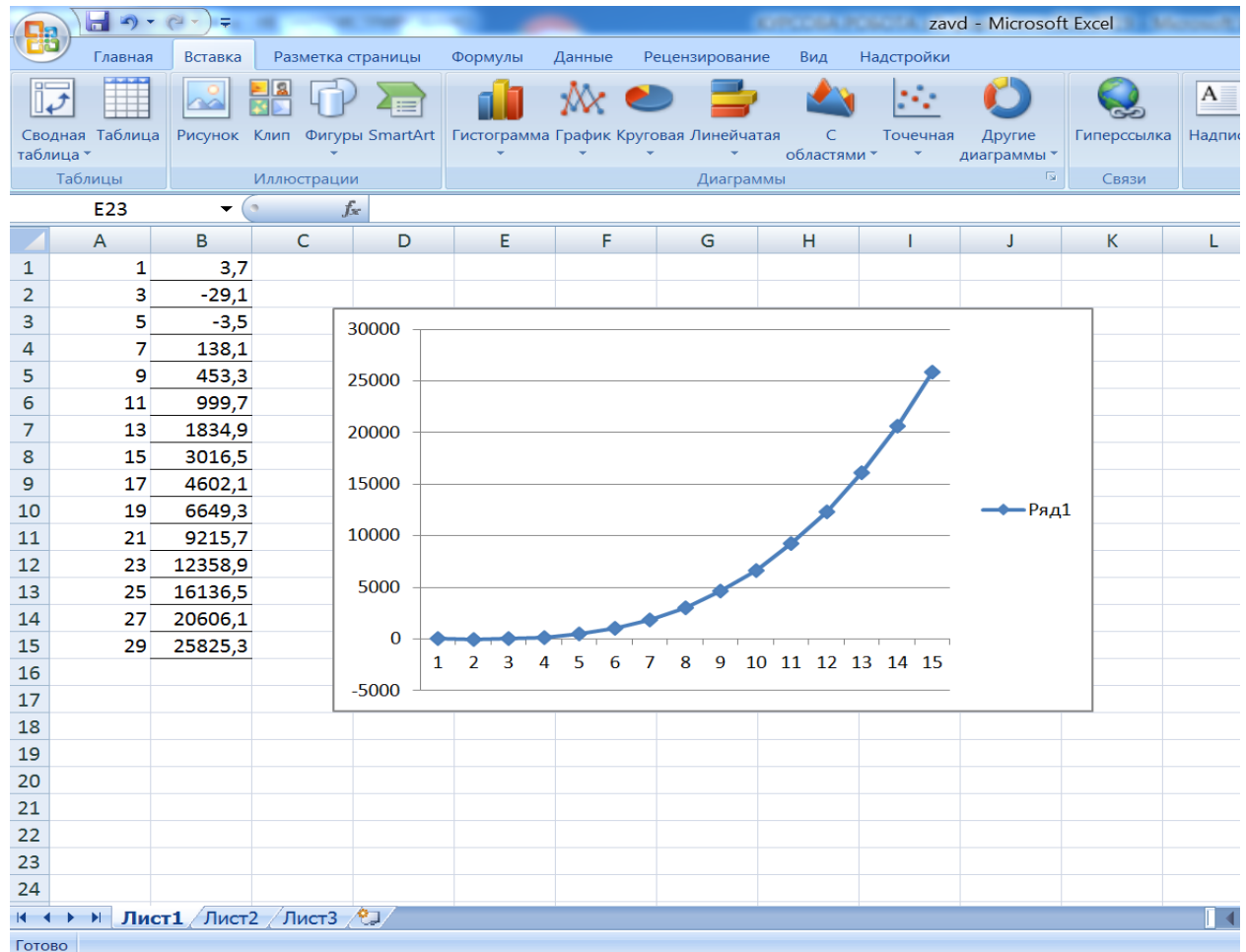


Рис.2. Створена таблиця даних та побудований графік функції  $y(x)$  у Excel

4.1.8 Очистити лист у Excel де побудовано графік функції  $y(x)$  (Рис.3);

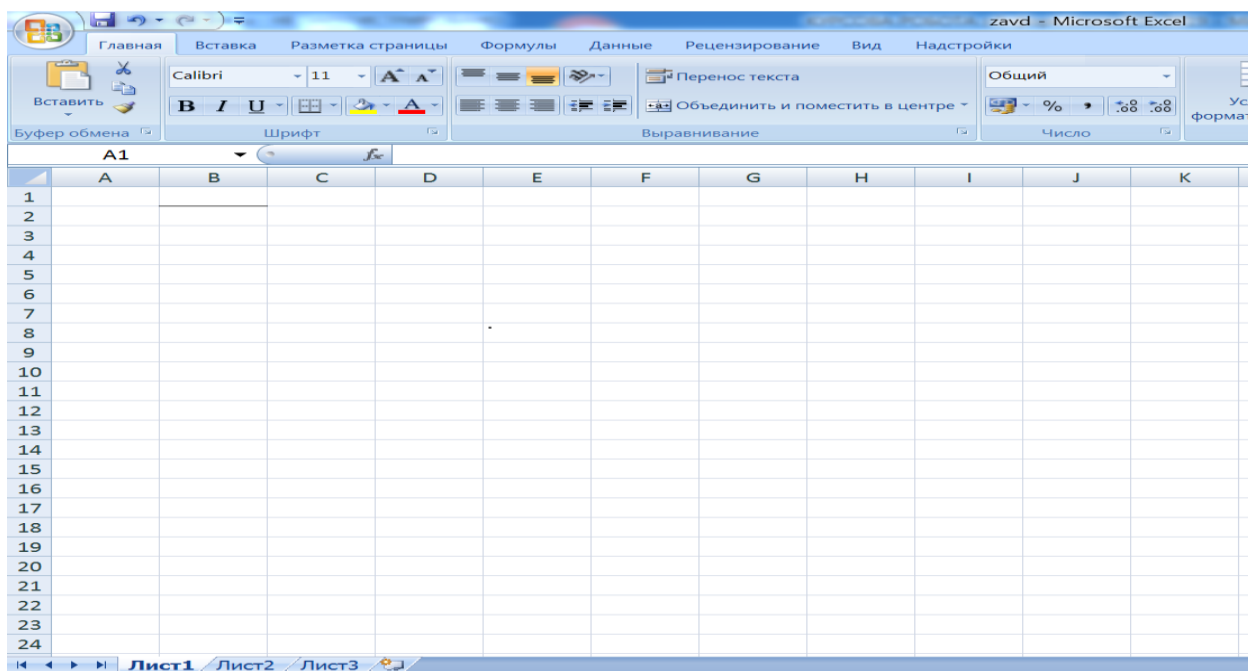


Рис.3. Лист у Excel після видалення таблиці даних та побудованого графіка функції  $y(x)$

4.1.9 Зайти у VBA для перегляду записаного макросу з побудови графіка функції  $y(x)$  у Excel, натиснувши правою кнопкою миші знизу ліворуч листа у Excel на Лист 1 та обравши в контекстному меню «Исходный текст»; отримаємо інтерфейс VBA без макроса (програми) (Рис.4);

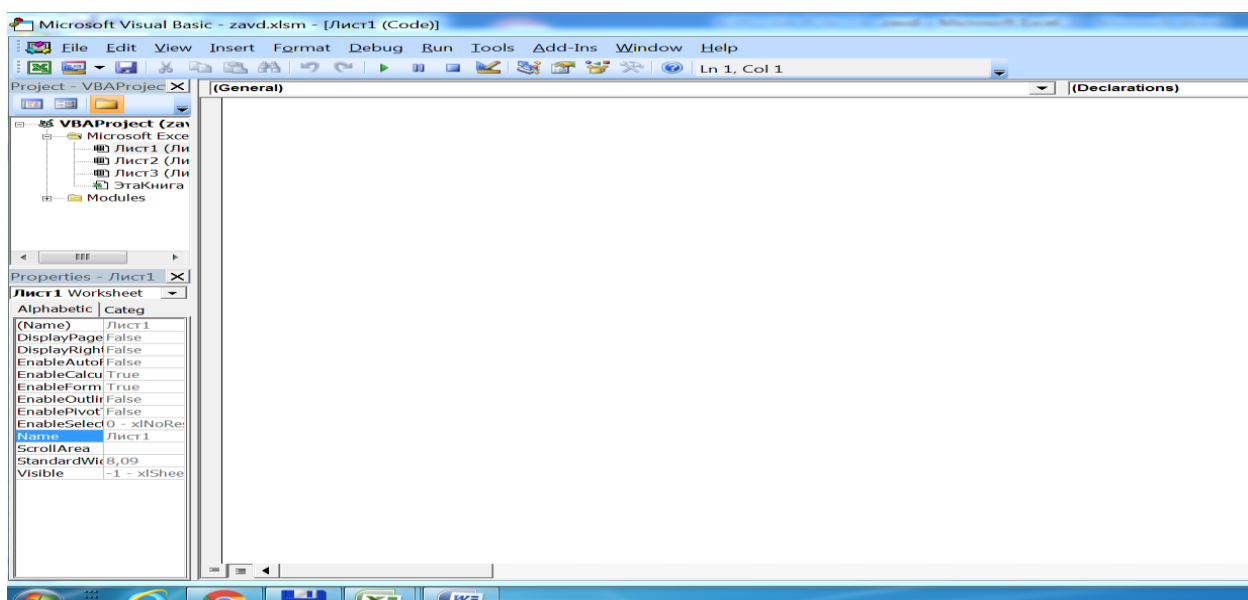


Рис.4. Інтерфейс VBA без програми

4.1.10. Натискаємо ліворуч у меню VBA у зоні проектів VBAProject у відкритій папці Modules на файл Modul1 і отримуємо у зоні VBA для написання програм створений макрос з побудови графіка функції  $y(x)$  у Excel макрорекордером VBA (Рис.5);

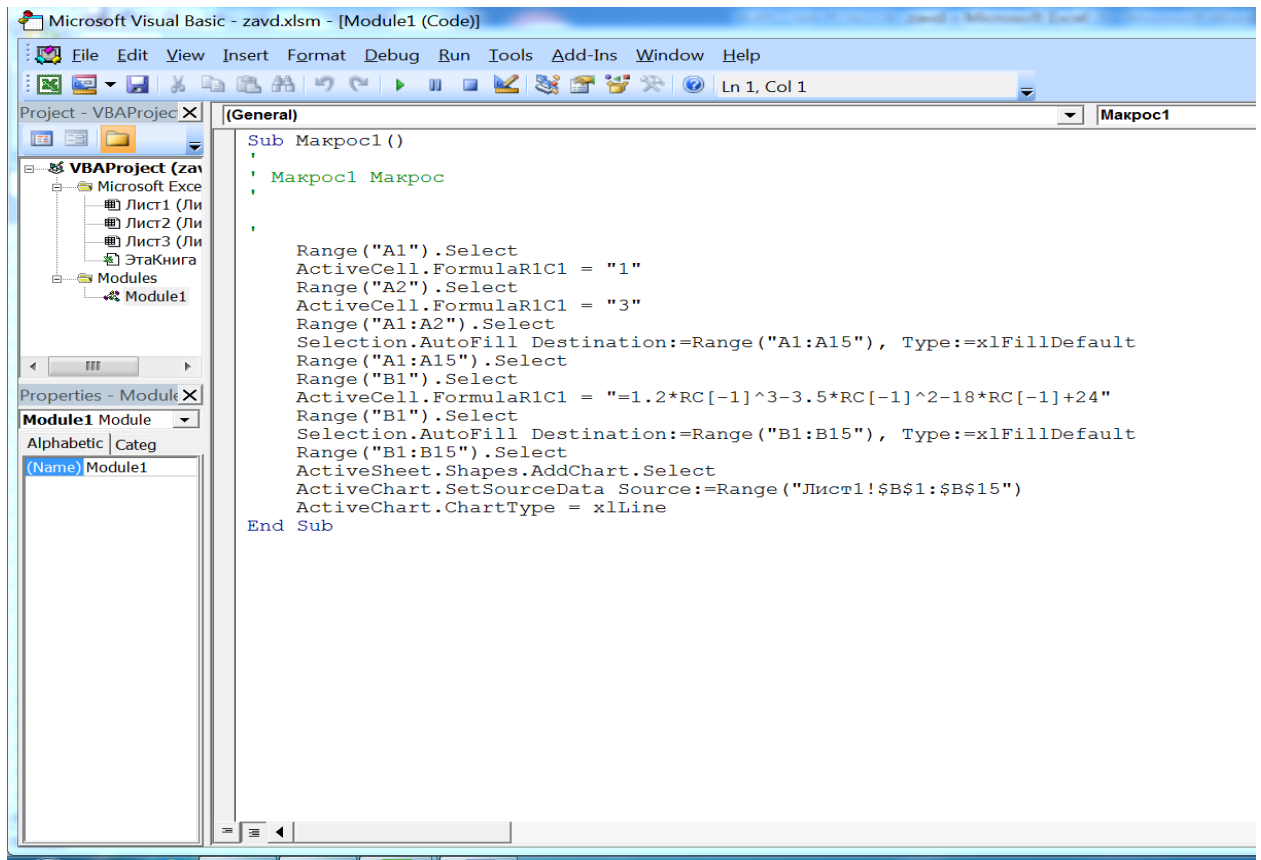


Рис.5. Створений макрос з побудови графіка функції  $y(x)$  у Excel

4.1.11 Скопіювати отриманий макрос (програму) у пояснювальну записку курсової роботи (Рис.6) і описати дії згідно п.п 4.1.1-4.1.10 щодо створення макроса макрорекордером VBA додатка Excel з побудови графіка функції  $y(x)$  за варіантом завдання курсової роботи;

4.1.12. До початку запуску макросу перевірити у пустому Листі 1 у Excel щоб була віділена комірка A1 та натиснути під стрічкою команд і кнопок Excel поруч з «Запуск макросов отключен» на «Параметри» і обрати з запропонованого «Включить это содержимое». Запустити написаний макрос на виконання, натиснувши лівою кнопкою миші у меню команд VBA на команду Run, а потім у контекстному меню натиснути на Run Sub та

перевірити та зробити скрін-шот результату виконання макроса у Excel з побудови графіка функції  $y(x)$  (Рис. 6);

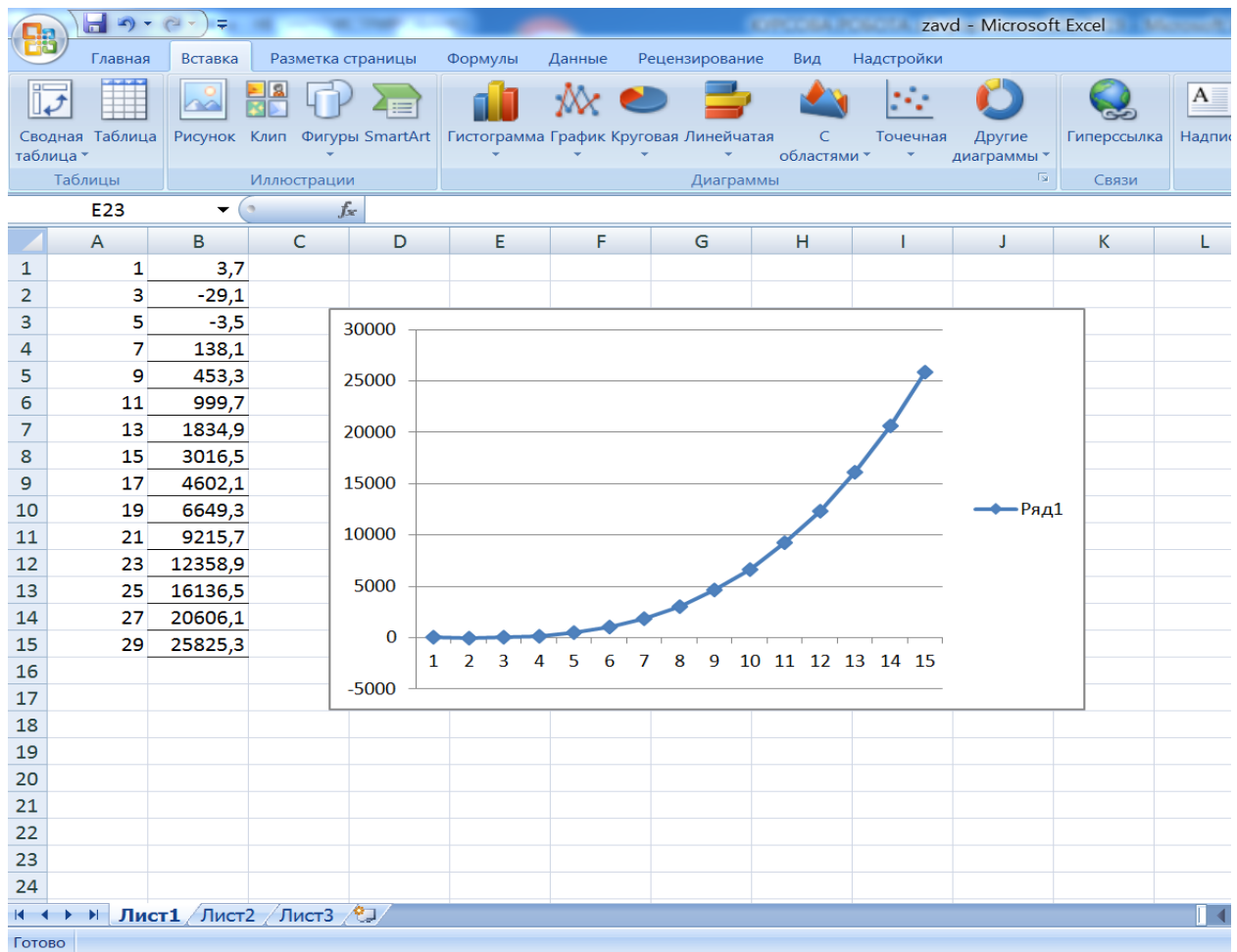


Рис. 6. Результат виконання макросу у Excel з побудови графіка функції  $y(x)$

4.1.13. Порівняти побудовані графіки функції з побудови графіка функції  $y(x)$ , які створені власноруч користувачем у Excel (Рис. 2) та автоматично макросом VBA (Рис.6); зробити висновки про ідентичність отриманих графіків.

## 4.2 Модифікація програмного коду макросу з побудови графіка функції за завданням курсової роботи

Для модифікації програмного коду макросу з побудови графіка потрібно:



4.2.1 Ввести перше та друге значення аргументу в комірки A1, A2 через діалогове віконце;

4.2.2 Через діалогове віконце ввести кількість точок для обчислення функції згідно варіанту завдання з курсової роботи (наприклад, для 15 значень аргументу функції);

4.2.3 Ввести коефіцієнти рівняння у вигляді змінних (a, b, c, d), значення яких вводяться через діалогове віконце (за замовчуванням – коефіцієнти у функції прийняти згідно варіанту за завданням до курсової роботи);

4.2.4 Показати користувачеві через діалогове віконце вигляд отриманого полінома за завданням з курсової роботи (наприклад, у виді: « $1.2x^3 - 3.5x^2 - 18x + 24$ »);

Після виправлень програмного коду у VBA згідно п.п. 4.2.1-4.2.4 отримаємо модифікований макрос з побудови графіка функції.

```
Sub zavd 1()
```

```
Cells.Clear
```

```
Dim a1, b1, n As Integer
```

```
' Dim a, b, c, d As Single
```

```
a1 = InputBox("A1=", "Введіть значення комірки A1", 1)
```

```
b1 = InputBox("A2=", " Введіть значення комірки A2", 3)
```

```
n = InputBox("n=", "Кількість значень для графіку", 15)
```

```
a = InputBox("a=", "Значення коефіцієнта a", "1.2")
```

```
b = InputBox("b=", " Значення коефіцієнта b", "-3.5")
```

```
c = InputBox("c=", " Значення коефіцієнта c", "-18")
```

```
d = InputBox("d=", " Значення коефіцієнта d", "24")
```

```
ActiveCell.FormulaR1C1 = a1
```

```
Range("A2").Select
```

```
ActiveCell.FormulaR1C1 = b1
```

```
Range("A1:A2").Select
```

```

Selection.AutoFill Destination:=Range(Cells(1, 1), Cells(n, 1)),
Type:=xlFillDefault
Range("B1").Select
ActiveCell.FormulaR1C1 = "=" & a & "*RC[-1]^3-" & b & "*RC[-1]^2+" & c
& "*RC[-1] +" & d & "="
Result = MsgBox("'" & a & "*x^3-" & b & "*x^2+" & c & "*x+" & d & "=",
vbOKCancel, "Підтвердіть запит:")
Range("B1").Select
Selection.AutoFill Destination:=Range(Cells(1, 2), Cells(n, 2)),
Type:=xlFillDefault
ActiveSheet.Shapes.AddChart.Select
ActiveChart.ChartType = xlXYScatterSmooth
ActiveChart.SetSourceData Source:=Range(Cells(1, 1), Cells(n, 2))
End Sub

```

4.2.5 Запустити модифікований макрос на виконання в VBA, для чого натиснути лівою кнопкою миші у меню команд VBA на команду Run, а потім у контекстному меню натиснути на Run Sub та перевірити; зробити скріншот результату виконання макроса у Excel з побудови графіка функції  $y(x)$ ;

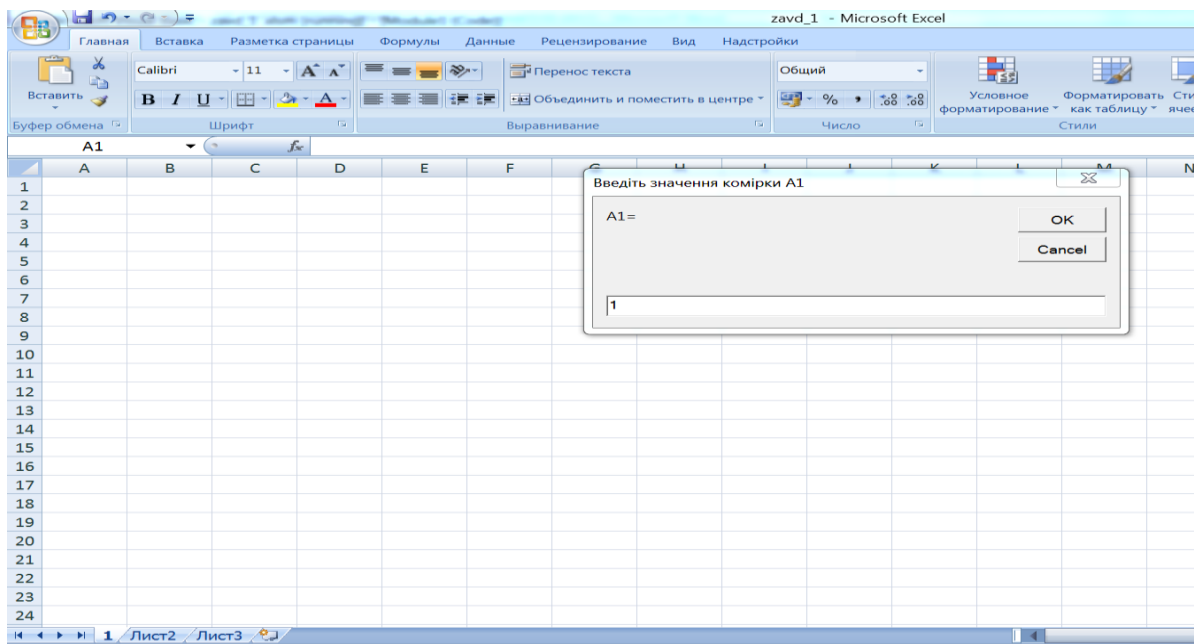


Рис. 7. Ввід початкового значення  $x$  у комірку A1 через віконце

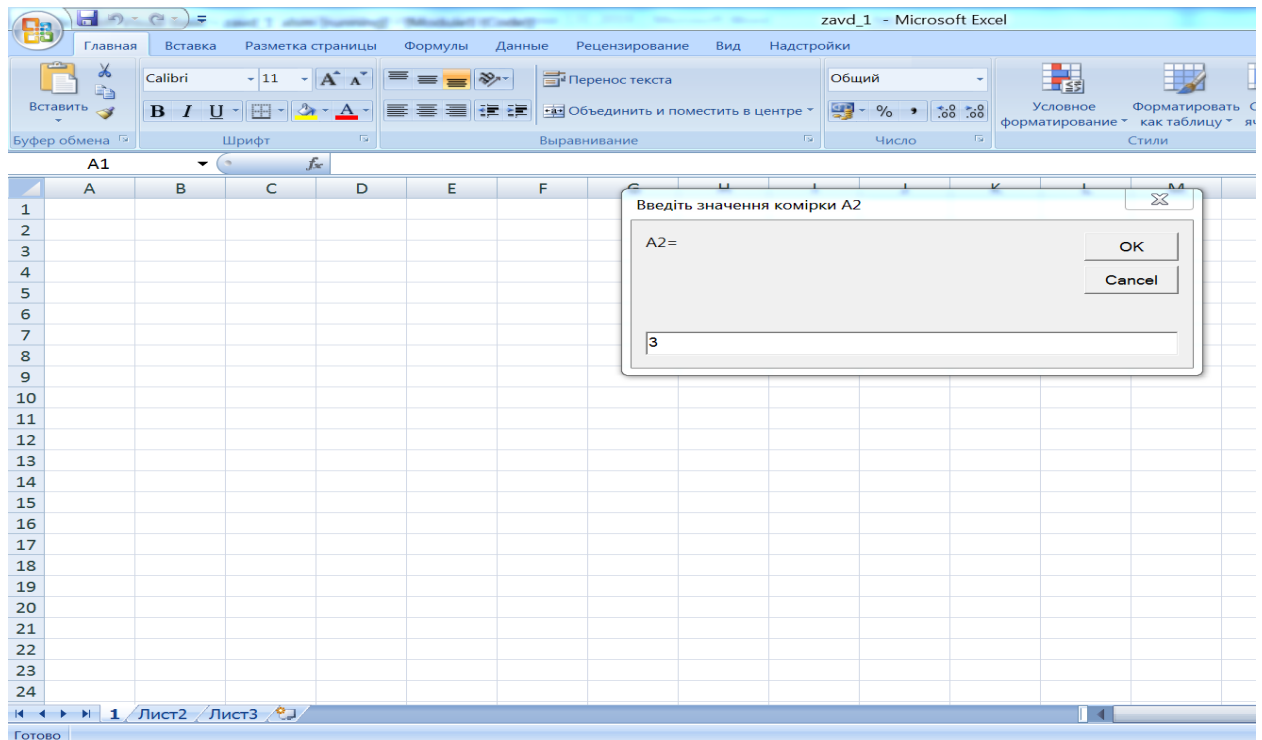


Рис8. Ввід другого значення  $x$  у комірку A2 через віконце

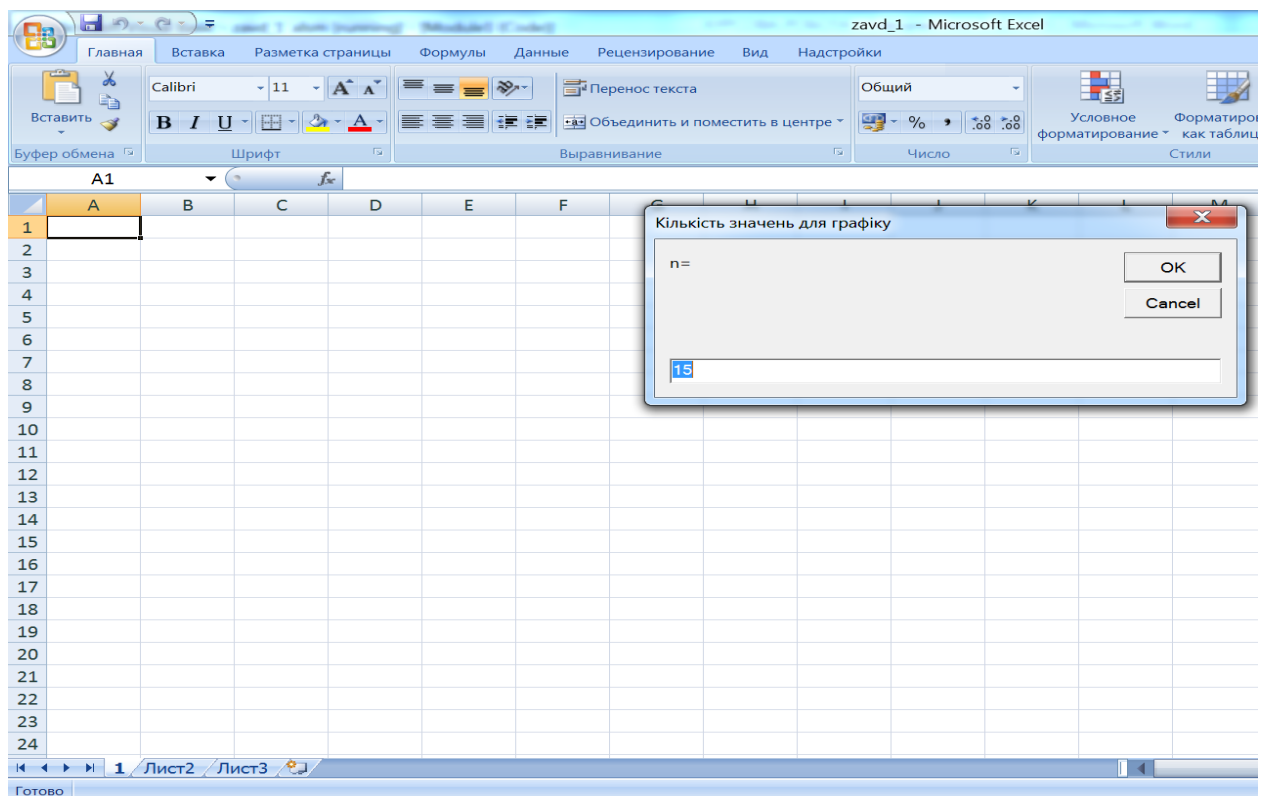


Рис.9. Ввід кількості значень  $n$  для побудови графіку функції  $y(x)$  через віконце

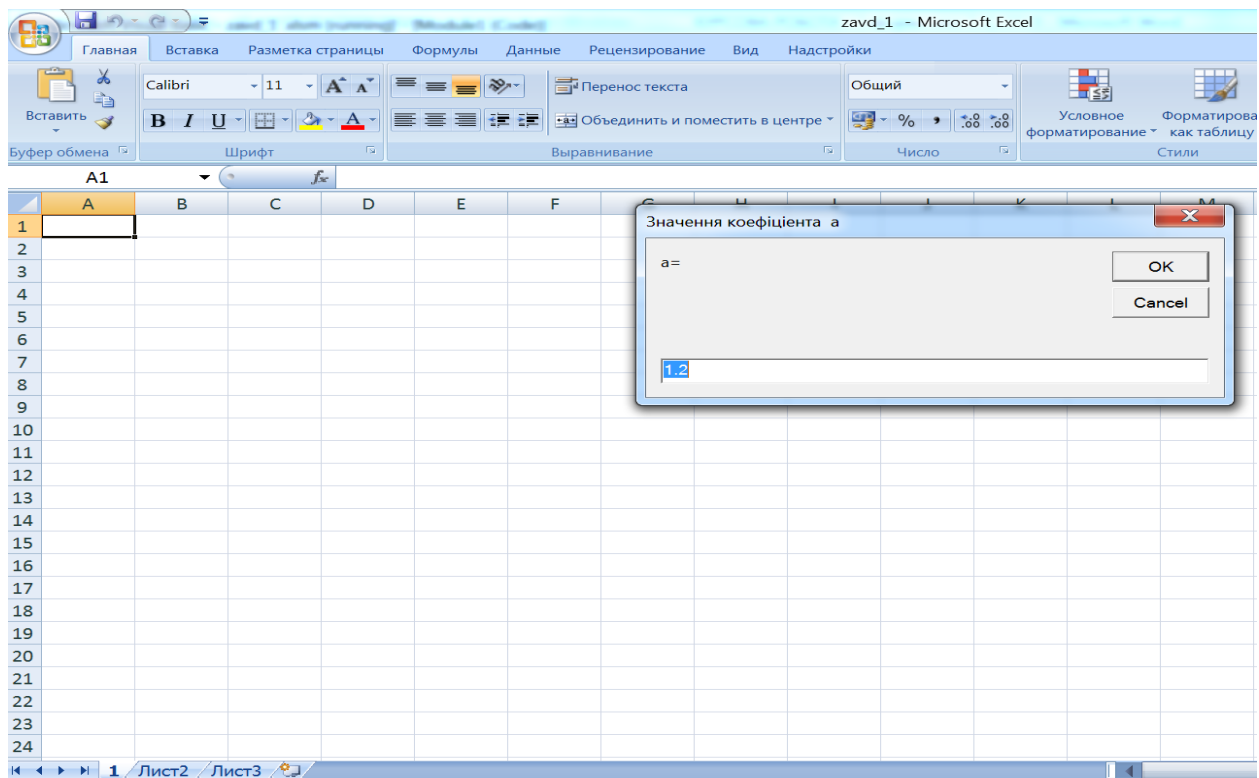


Рис. 10. Ввід числового значення коефіцієнта  $a$  через віконце

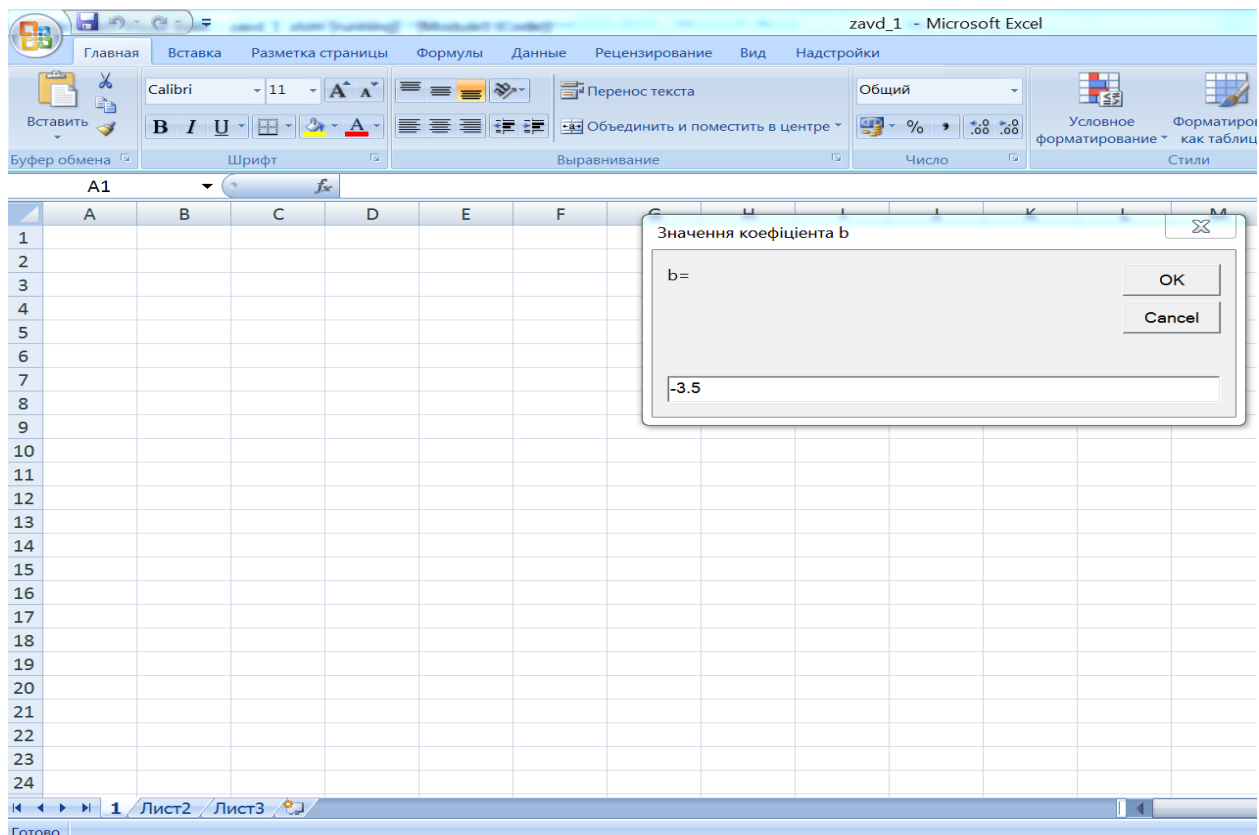


Рис. 11. Ввід числового значення коефіцієнта  $b$  через віконце

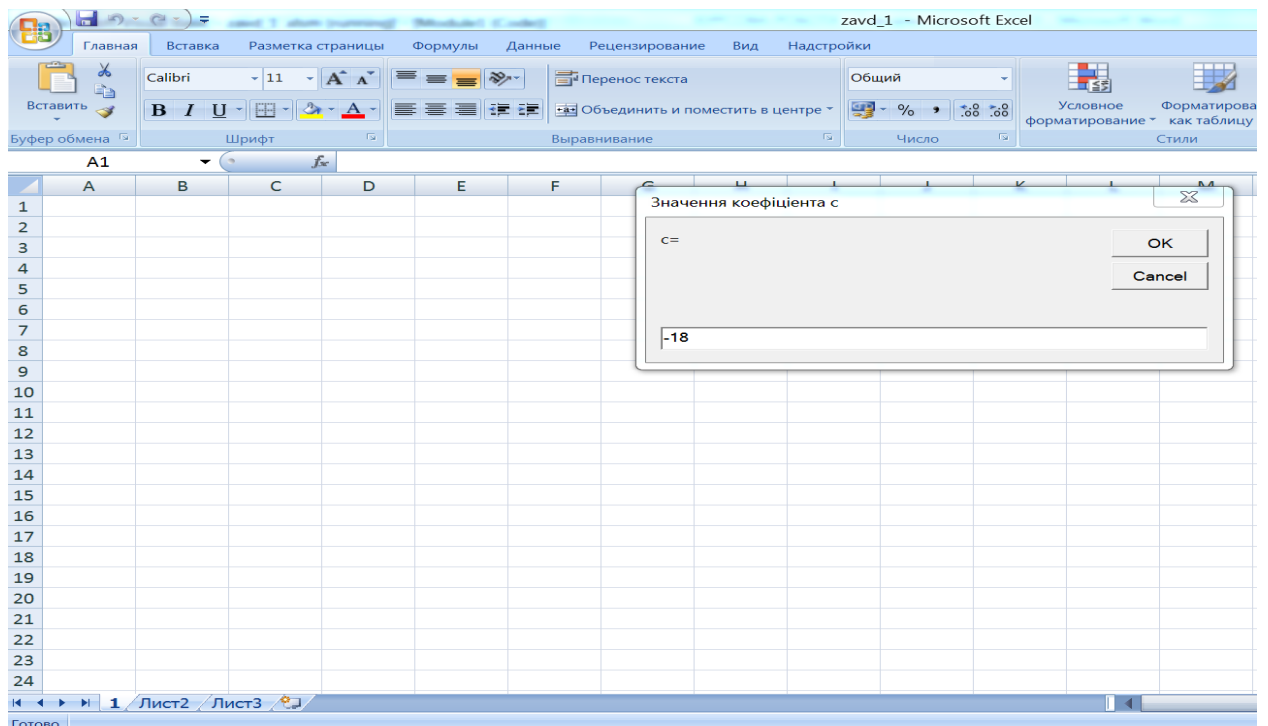


Рис. 12. Ввід числового значення коефіцієнта c через віконце

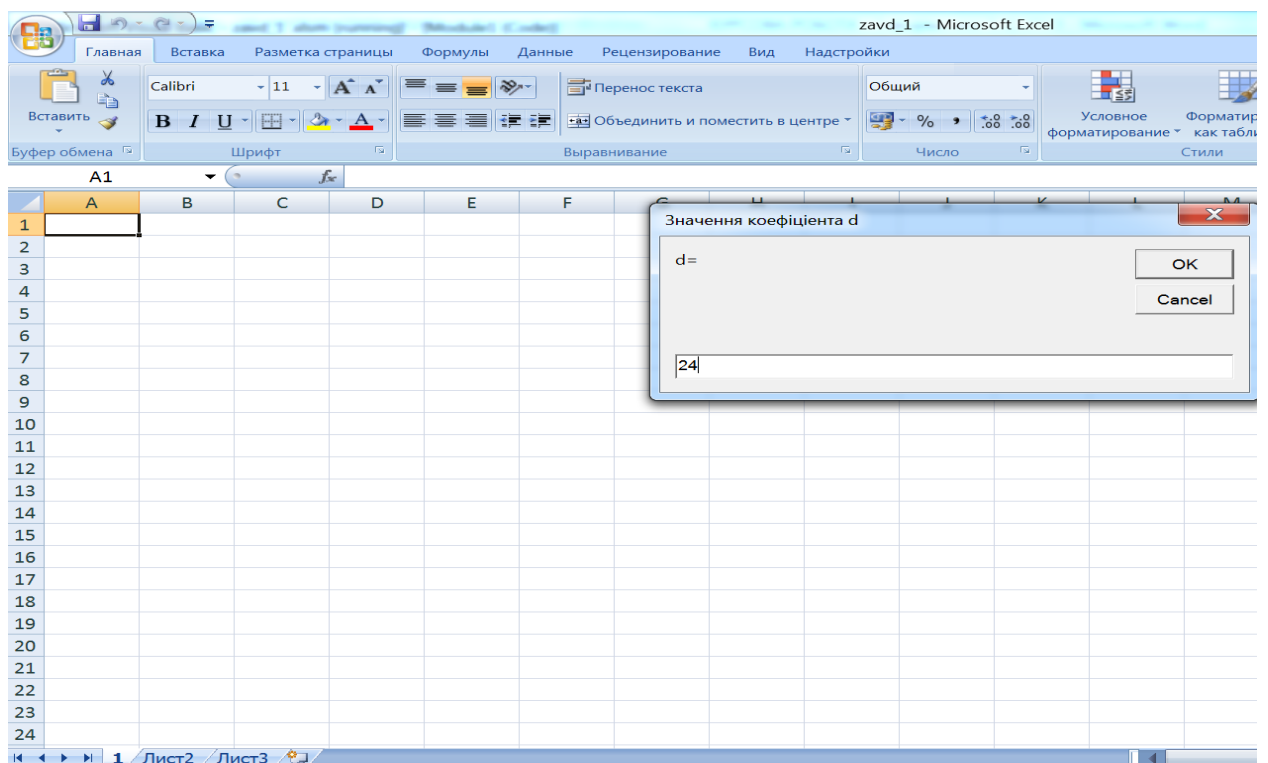


Рис.13. Ввід числового значення коефіцієнта d через віконце

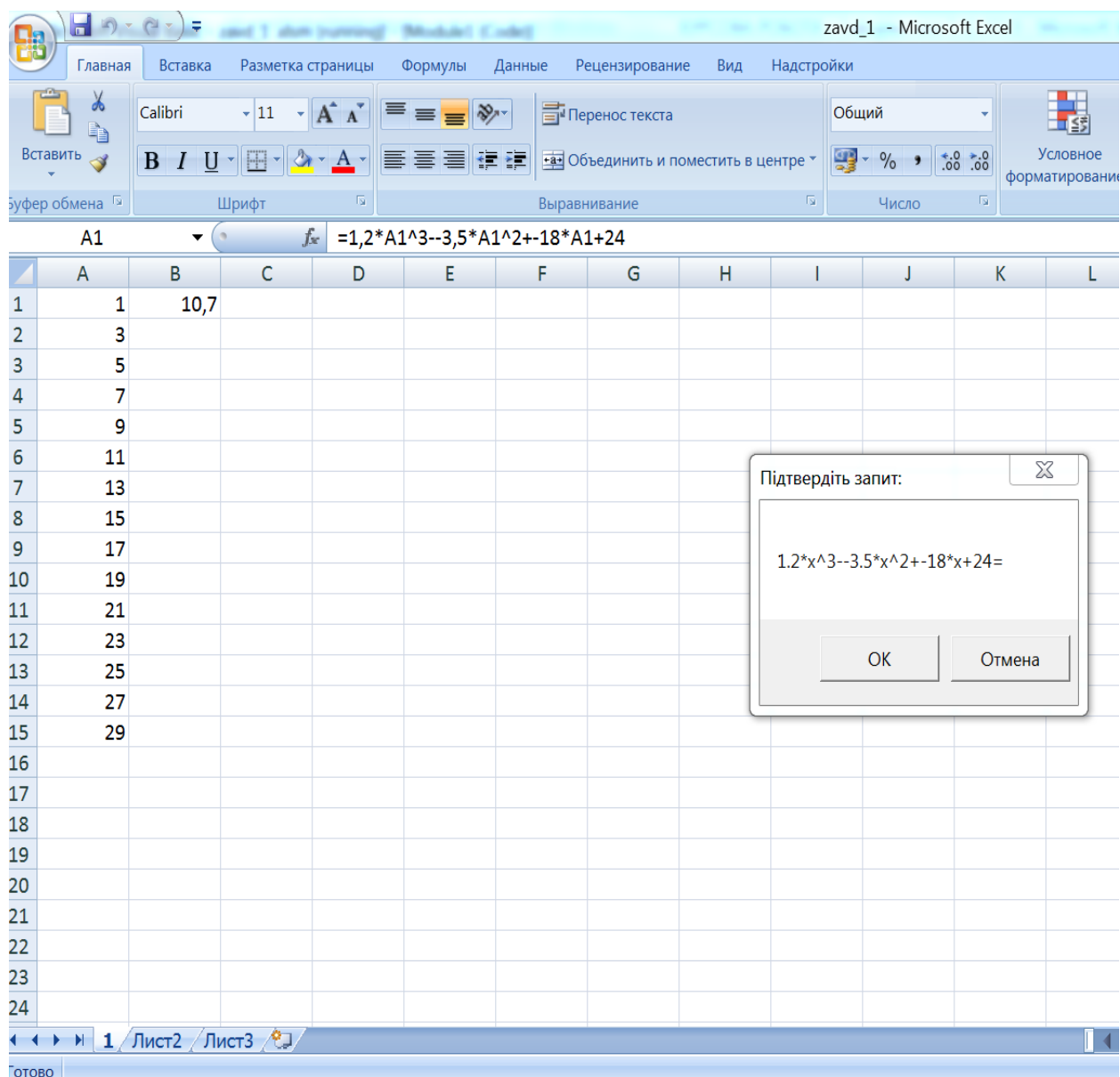


Рис.14. Вивід рівняння функції  $y(x)$  через віконце

4.2.6. Представити скрін-шот модифікованого макроса у вікні VBA з побудови графіка функції  $y(x)$  у Excel за варіантом завдання курсової роботи (Рис.15);

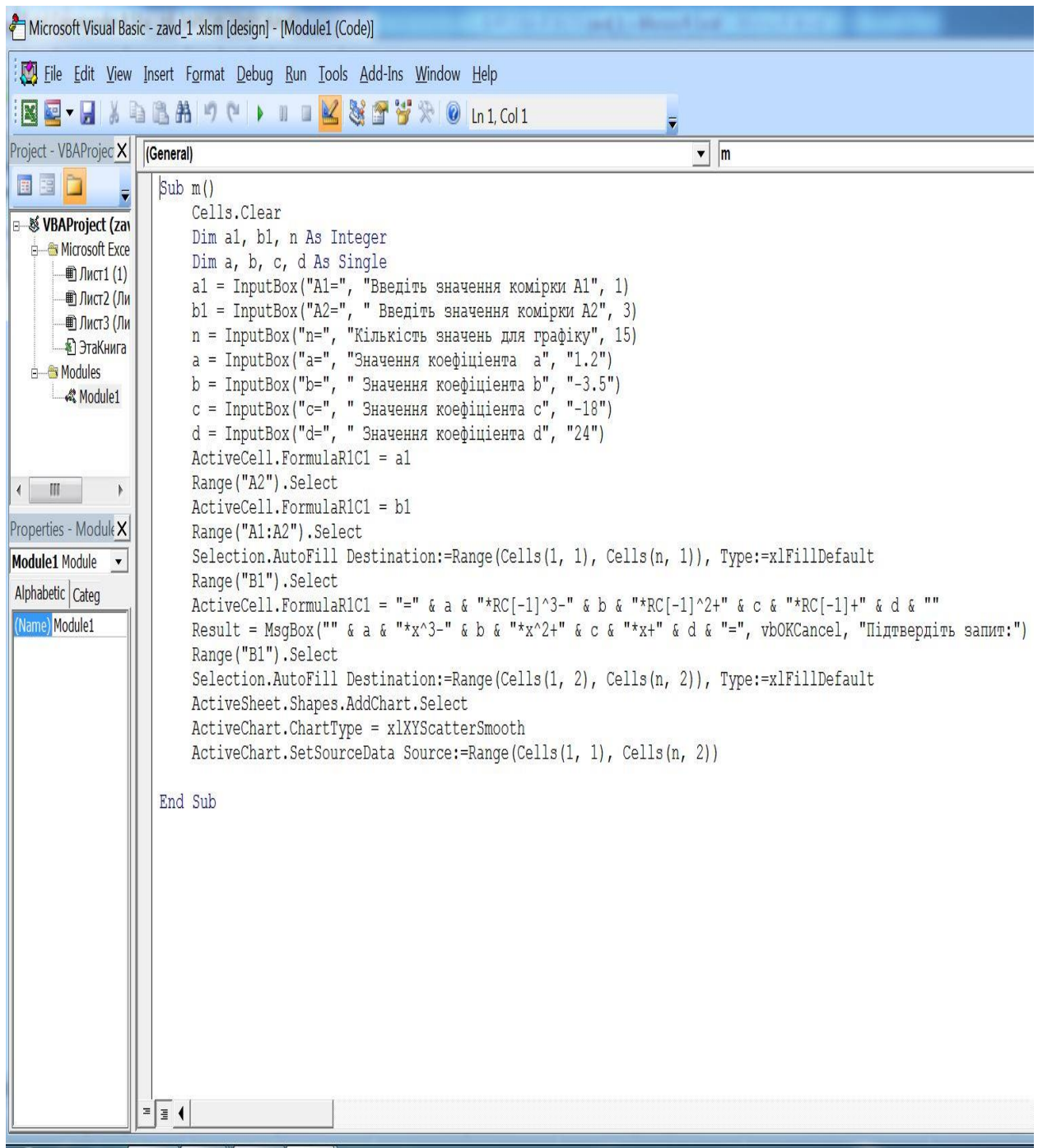


Рис. 15. Скрін-шот модифікованого макроса з побудови графіка функції  $y(x)$  у Excel

4.2.7. Закрити вікно VBA з модифікованим макросом з побудови графіка функції  $y(x)$  у Excel за варіантом завдання курсової роботи і перевірити у Листі 1 Excel результат виконання модифікованого макроса (Рис. 16);

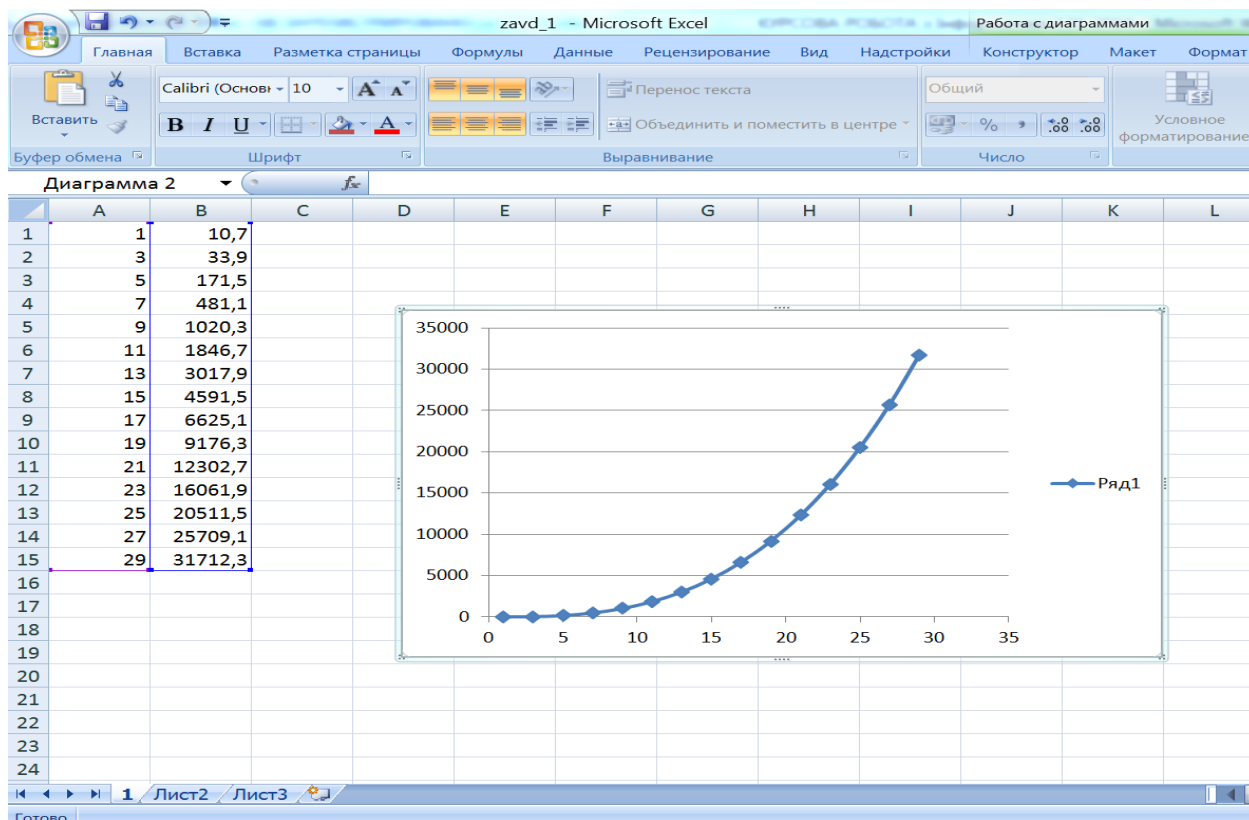


Рис. 16. Графік функції  $y(x)$  автоматично створений макросом VBA

4.2.8. Описати дії щодо створення модифікованого макроса в VBA згідно п.п. 4.2.1-4.2.5 і представити скрін-шот модифікованого макроса.

4.2.9. Порівняти побудовані графіки функції з побудови графіка функції  $y(x)$ , які створені власноруч користувачем у Excel (Рис. 2) та автоматично макросом VBA (Рис.16); зробити висновки про ідентичність отриманих графіків.

### 4.3. Складання блок-схеми для написання програми в VBA з побудови графіка функції за завданням курсової роботи

Скласти блок-схему для написання програми в VBA з побудови графіка функції за варіантом завдання курсової роботи у текстовому редакторі Word. Навести скрін-шот блок-схеми програми в VBA з побудови графіка функції



за варіантом завдання курсової роботи на основі використання необхідних блоків для алгоритму програми (Табл.2).

#### **4.4. Створення програми в VBA з побудови графіка функції за завданням курсової роботи**

Повні можливості програмування в Office розкриваються при використанні редактора Visual Basic. Для входу в нього в меню Сервіс | Макрос вибрати «Редактор Visual Basic», або натиснути клавіші <Alt> + <F11>.

У верхньому лівому кутку зазвичай відкрито вікно провідника проекту. Якщо воно випадково було закрито, то викликати його можна, натиснувши клавіші <Ctrl> + <R>.

Самий верхній рівень – це проект (Project), якому відповідає робоча книга Excel.

Якщо вам потрібно створити програму вручну, а макросів в даному документі ще немає, то потрібно клацнути правою кнопкою миші по вузлу проекту (рядку, виділеної напівжирним шрифтом) і в контекстному меню вибрати команду Insert → Module. У проекті буде створено новий модуль і відкритий у вікні редактора коду. Починатися програма повинна з оголошення **Sub назва ()**. Закінчується програма командою **End Sub**.

Створення програми з побудови графіка функції за завданням курсової роботи виконується в VBA за допомогою потрібних операторів (вводу вихідних даних в комірку, виводу результатів розрахунку функцій в віконце), починаючи з оператора **Sub r()** і завершуючи **End Sub**.

Результат створення програми з побудови графіка функції за завданням курсової роботи в VBA представити у вигляді скрін-шота.

Приклад фрагменту програми:

```

Sub r()
    Dim n As Integer
    Dim a, b, c, d As Single
    ... Cells (1, 1) = a1
    .
    .
    .
    Result = MsgBox("'" & a & "*x^3-" & b & "*x^2+" & c & "*x+" & d &
    "=", vbOKCancel, "Підтвердіть запит:")
End Sub

```

Описати дії щодо написання програми в VBA з побудови графіка функції за завданням курсової роботи з використанням потрібних операторів і зробити скрін-шот написаної програми.

Зробити висновки до курсової роботи з дисципліни «Інформатика», в яких проаналізувати збіжність результатів виконання завдання з курсової роботи трьома способами.

## **КОНТРОЛЬНІ ПИТАННЯ**

1. Як записати макрос у вкладці Microsoft Office Excel за допомогою макрорекордера VBA?
2. Як увійти в редактор VBA у вкладці Microsoft Office Excel?
3. Які основні типи даних існують і застосовують у програмах у VBA у вкладці Microsoft Office Excel?
4. Які основні операції можна проводити зі строковими змінними у програмі у VBA у вкладці Microsoft Office Excel?
5. Якими шляхами можна ввести вихідні дані згідно завдання у програмі у VBA користувачу?
6. Якими шляхами можна вивести результати розрахунку рівняння функції у програмі у VBA користувачу?

**ВАРІАНТИ ЗАВДАНЬ ДО КУРСОВОЇ РОБОТИ З ДИСЦИПЛІНИ  
«ІНФОРМАТИКА»**

1. Створити макрос з побудови графіка функції  $y(x)$ ; 2. Модифікувати створений макрос; 3. Розробити алгоритм і програму знаходження значень функції  $y(x)$  у VBA за варіантом:

1.  $y(x) = 3.51 \cdot x^3 - 14.2 \cdot x^2 + 18 \cdot x + 53$ ;
2.  $y(x) = 0.02 \cdot x^3 - 1.85 \cdot x^2 + 31 \cdot x + 15$ ;
3.  $y(x) = 0.21 \cdot x^3 - 1.31 \cdot x^2 + 2.1 \cdot x - 41$ ;
4.  $y(x) = 0.69 \cdot x^3 - 0.53 \cdot x^2 + 15 \cdot x + 21$ ;
5.  $y(x) = 1.9 \cdot x^3 - 3.7 \cdot x^2 + 19 \cdot x - 17$ ;
6.  $y(x) = 0.02 \cdot x^3 - 1.5 \cdot x^2 + 23 \cdot x + 10$ ;
7.  $y(x) = 0.15 \cdot x^3 - 1.8 \cdot x^2 + 3.6 \cdot x + 10.5$ ;
8.  $y(x) = 0.31 \cdot x^3 + 1.1 \cdot x^2 + 2.3 \cdot x + 10$ ;
9.  $y(x) = 0.14 \cdot x^3 - 0.89 \cdot x^2 + 5 \cdot x + 90$ ;
10.  $y(x) = 2.5 \cdot x^3 - 1.8 \cdot x^2 + 2 \cdot x + 16$ ;
11.  $y(x) = 2.81 \cdot x^3 - 12.7 \cdot x^2 + 25 \cdot x + 71$ ;
12.  $y(x) = 0.09 \cdot x^3 - 2.75 \cdot x^2 + 30 \cdot x + 26$ ;
13.  $y(x) = 0.28 \cdot x^3 - 4.33 \cdot x^2 + 3.1 \cdot x - 61$ ;
14.  $y(x) = 0.43 \cdot x^3 - 0.57 \cdot x^2 + 12 \cdot x + 14$ ;
15.  $y(x) = 3.8 \cdot x^3 - 4.8 \cdot x^2 + 14 \cdot x - 17$ ;
16.  $y(x) = 0.06 \cdot x^3 - 1.6 \cdot x^2 + 21 \cdot x + 16$ ;
17.  $y(x) = 0.2 \cdot x^3 - 1.5 \cdot x^2 + 5 \cdot x + 14.5$ ;
18.  $y(x) = 0.31 \cdot x^3 + 1.1 \cdot x^2 + 2.3 \cdot x + 10$ ;
19.  $y(x) = 0.1 \cdot x^3 - 0.8 \cdot x^2 + 3 \cdot x + 100$ ;
20.  $y(x) = 1.5 \cdot x^3 - 0.8 \cdot x^2 + 5 \cdot x + 15$ ;
21.  $y(x) = 1.51 \cdot x^3 - 10.2 \cdot x^2 + 28 \cdot x + 61$ ;
22.  $y(x) = 0.05 \cdot x^3 - 2.82 \cdot x^2 + 40 \cdot x + 25$ ;
23.  $y(x) = 0.21 \cdot x^3 - 1.31 \cdot x^2 + 2.1 \cdot x - 41$ ;
24.  $y(x) = 0.38 \cdot x^3 - 0.34 \cdot x^2 + 18 \cdot x + 24$ ;
25.  $y(x) = 1.8 \cdot x^3 - 4.7 \cdot x^2 + 12 \cdot x - 18$ .

## **СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ**

1. Інформатика: Excel та Basic for Application: навч. посіб. / Лопотко О.В. Рекомендовано МОН України. - К. : Вид-во «Каравела», 2018. - 272 с.
2. Білан Б.С., Карпович І.М. Інформатика та інформаційні технології: навч. посіб. – Рівне: НУВГП, 2010. – 197 с.
3. Гарберг Г.З. Основы программирования на VB и VBA в Excel 2007 / Г.З. Гарберг– М.: Солон-Пресс, 2008.- 192 с.
4. Згуровський М.З., Коваленко І.І., Міхайленко В.М. Вступ до комп'ютерних інформаційних технологій: навч.посіб. – К.: Вид-во Європ. ун-ту (фінанси, інформ. системи, менеджм. і бізнес), 2000.- 265 с.
5. Інформатика та комп'ютерна техніка: навч. посіб. / М.Є. Рогоза. - К. : ВЦ «Академія», 2006. - 368 с.
6. Інформатика та комп'ютерна техніка: посіб. / Л.М. Дибкова. - К. : ВЦ «Академія», 2002. - 320 с.
7. Басков Н.Н. Visual Basic for Applications: учебн. практикум / Н.Н.Басков, А.Р. Лебединская. – Ростов на Дону: РГТЭУ, 2008. – 83 с.
8. Караванова Т.П. Інформатика. Збірник вправ та задач алгоритмізації та програмування: навч. посіб. / Т.П. Караванова. - Шепетівка: ПП «Шестопапов С.А.», 2017. - 152 с.

## **СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ**

1. Грошев А.С. Информатика: учебник для вузов / А.С. Грошев. - Архангельск, Арханг.гос.техн.ун-т, 2010.- 470 с.
2. Слепцова Л.Д. Программирование на VBA в Microsoft Office 2007. Самоучитель / Л.Д. Слепцов. - М., 2009. – 432 с.
3. Лук'янова В.В. Комп'ютерний аналіз даних: посібн. / В.В. Лук'янова. – К.: Видавничий центр „Академія”, 2003. – 344с.

ДОДАТОК 1

## Зразок титульного листа

Міністерство освіти і науки України  
Національний технічний університет України  
«Київський політехнічний інститут  
імені Ігоря Сікорського»

Інженерно-хімічний факультет

Кафедра хімічного, полімерного і силікатного машинобудування

# ІНФОРМАТИКА

## КУРСОВА РОБОТА

Прийняв:  
Казак  
Ірина Олександрівна  
Захищено з оцінкою

---

Виконав:  
Іванов Євген Ігорович  
студент групи ЛУ-  
Варіант №.....

Київ  
201

Електронне мережне навчальне видання

Казак Ірина Олександрівна

# **ІНФОРМАТИКА**

## **КУРСОВА РОБОТА**

*для підготовки бакалаврів, які навчаються за спеціальністю  
131 – «Прикладна механіка», освітньою програмою «Інжиніринг паковань  
та пакувального обладнання»*

Комп'ютерна правка та верстка – *авторські*